Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (Currently Amended) A method to manage packet fragmentation for address translation, comprising:

receiving a plurality of packet fragments for a packet having a first address, wherein each packet fragment includes a packet fragment header having a packet offset value, said offset value representing a position from a starting position to an ending position of said packet;

translating said first address into a second address without reassembling said packet fragments into said packet;

determining whether all packet fragments for said packet have been received by indexing said offset value by position in a verification table to identify any missing positions between said starting position and said ending position; and

sending said packet fragments using said second address.

2. (Previously Presented) The method of claim 1, wherein said translating comprises:

identifying said packet fragment having a packet header, with said packet header having a packet identifier, translation information and a packet length;

retrieving translation information from said packet header; and

translating said first address into said second address using said translation information.

3. (Original) The method of claim 2, wherein said translation information comprises a port number.

4. (Previously Presented) The method of claim 2, wherein each of said packet fragment includes said packet fragment header having said packet identifier and a more bit, and said determining comprises:

storing each packet fragment having said packet identifier and said more bits set to predetermined values.

5. (Previously Presented) The method of claim 1, wherein each offset value represents a position for said packet fragment in said packet, and said determining whether all packet fragments for said packet have been received using said offset values comprises:

collecting said offset values;

retrieving said packet length; and

determining whether all positions for said packet are filled by said collected offset values using said packet length.

6. (Original) The method of claim 2, wherein each packet fragment includes a packet fragment header having said packet identifier, a more bit and an offset value, and said determining comprises:

storing each packet fragment having said packet identifier and said offset value is a value other than zero; and

determining whether all packet fragments for said packet have been received using said offset values.

7. (Original) The method of claim 6, wherein each offset value represents a position for said packet fragment in said packet, and said determining whether all packet fragments for said packet have been received using said offset values comprises:

collecting said offset values;

retrieving said packet length; and

determining whether all positions are filled by said collected offset values using said packet length.

8. (Original) The method of claim 5, wherein each offset value represents a position in bytes divided by eight for said packet fragment in said packet.

9. (Original) The method of claim 7, wherein each offset value represents a position in bytes divided by eight for said packet fragment in said packet.

10. (Original) The method of claim 1, further comprising: detecting an occurrence of a terminating condition prior to receiving all of said packet fragments for said packet; and releasing said packet fragments in accordance with said detection.

11. (Currently Amended) A packet fragmentation manager to manage packet fragmentation for address translation, comprising:

a collection module to collect and store a plurality of packet fragments for a packet having a first address, wherein each packet fragment includes a packet fragment header having a packet offset value, said offset value representing a position from a starting position to an ending position of said packet;

a verification module to verify all packet fragments for said packet have been received; and

a translation module to retrieve translation information from one of said packet fragments and ~~translating~~ to translate said first address into a second address using said translation information, without reassembling said packet fragments into said packet, wherein said verification module is to determine whether all packet fragments for said packet have been received by indexing said offset value by position in a verification table to identify any missing positions between said starting position and said ending position.

12. (Previously Presented) The packet fragmentation manager of claim 11, further comprising a communication module to send said packet fragments to said second address.

13. (Currently Amended) A system to manage packet fragmentation for an address translation device, comprising:

a source node to send packet fragments for a packet having a first address; and

an intermediate node to receive said packet fragments and to translate said first address to a second address without reassembling said packet fragments into said packet;

wherein said intermediate node is further adapted to index offset values from each packet fragment in a verification table to determine whether all packet fragments for said packet have been received, each of said offset values representing a position from a starting position to an ending position of said packet, said intermediate node to index said offset values by position in said verification table to identify any missing positions between said starting position and said ending position.

14. (Original) The system of claim 13, further comprising a destination node having said second address to receive said packet fragments and reassemble said packet fragments into said packet.

15. (Currently Amended) A system to manage packet fragmentation for an address translation device, comprising:

a computer platform adapted to manage packet fragmentation;

said platform being further adapted to receive a plurality of packet fragments for a packet having a first address, translate the first address into a second address without reassembling said packet fragments into said packet, and send said packet fragments using said second address;

wherein said platform is further adapted to index offset values from each packet fragment in a verification table to determine whether all packet fragments for said packet have been received, each of said offset values representing a position from a starting position to an ending position of said packet, said intermediate node to index said offset values by position in said verification table to identify any missing positions between said starting position and said ending position.

16. (Original) The system of claim 15, wherein said platform is further adapted to perform said translation by identifying a packet fragment having a packet header, with said packet header having a packet identifier, translation information and a packet length, determining whether all packet fragments for said packet have been received, retrieving translation information from said packet header, and translating said first address into said second address using said translation information.

17. (Previously Presented) The system of claim 15, wherein said platform is further adapted to collect said offset values, retrieve a packet length for said packet, and determine whether all positions for said packet are filled by said collected offset values using said packet length.

18. (Currently Amended) An article comprising:

a storage medium;

said storage medium including stored instructions that, when executed by a processor, result in receiving a plurality of packet fragments for a packet having a first

address, translating said first address into a second address without reassembling said packet fragments into said packet, and sending said packet fragments using said second address;

wherein the stored instructions, when executed by a processor, further result in indexing offset values from each packet fragment in a verification table to determine whether all packet fragments for said packet have been received, each of said offset values representing a position from a starting position to an ending position of said packet, said intermediate node to index said offset values by position in said verification table to identify any missing positions between said starting position and said ending position.

19. (Original) The article of claim 18, wherein the stored instructions, when executed by a processor, further result in said translating by identifying a packet fragment having a packet header, with said packet header having a packet identifier, translation information and a packet length, determining whether all packet fragments for said packet have been received, retrieving translation information from said packet header, and translating said first address into said second address using said translation information.

20. (Previously Presented) The article of claim 19, wherein the stored instructions, when executed by a processor, further result in collecting said offset values, retrieving a packet length for said packet, and determining whether all positions for said packet are filled by said collected offset values using said packet length.

21. (Original) The article of claim 18, wherein the stored instructions, when executed by a processor, further result in detecting an occurrence of a terminating condition prior to receiving all of said packet fragments for said packet, and releasing said packet fragments in accordance with said detection.